

Tanja Henttunen

DOKUMENTOINTI PELIALAN YRITYKSESSÄ

DOKUMENTOINTI PELIALAN YRITYKSESSÄ

Tanja Henttunen
Opinnäytetyö
Kevät 2014
Ohjelmistokehityksen koulutusohjelma
Oulun seudun ammattikorkeakoulu

TIIVISTELMÄ

Oulun seudun ammattikorkeakoulu
Tietotekniikan koulutusohjelma, ohjelmistokehityksen suuntautumisvaihtoehto

Tekijä: Tanja Henttunen

Opinnäytetyön nimi: Dokumentointi pelialan yrityksessä

Työn ohjaaja: Tuomo Tikkanen

Työn valmistumislukukausi ja -vuosi: kevät 2014 Sivumäärä: 39

Opinnäytetyön aiheena oli dokumentointi pelialan yrityksessä, jossa selvitetään, miksi dokumentaatioita tarvitaan pelialalla ja mitä yleisimmin käytetyt dokumentaatiot sisältävät. Tarkoituksena oli selvittää, onko pelialan dokumentoinnissa yhteisiä sääntöjä ja mitä yleisimmät dokumentit sisältävät. Opinnäytetyössä käsitellään myös dokumentoinnin hyötyjä ja haittoja sekä hyvän dokumentoinnin piirteitä.

Opinnäytetyötä varten tietoa etsittiin useista eri lähteistä ja vertaamalla niitä keskenään löydettiin yhdenmukaisin tieto. Lähteet koostuvat useista erilaisista pelinkehitykseen keskittyviltä sivustoista ja pelialaan liittyvistä kirjoista. Lisäksi tekstissä on käytetty pohjana työkokemuksia kahdesta pelialan yrityksestä.

Johtopäätöksenä on, ettei pelialalla ole yhtenäistä standardia dokumentaation tekemiseen. Ohjelmistoalalla on totuttu tapaan, jossa tehdään vain muunnellut versiot tärkeimmistä dokumentaatioista, kuten pelisuunnitelmasta. Vaikka ei yhtenäistä dokumentoinnin standardia ei ole pelialalla, toimii opinnäytetyö uusi- en työntekijöiden perehdyttämisen perusteena pelialan tärkeimpiin käytettyihin dokumentaatioihin, dokumentaation hyötyihin ja haittoihin sekä hyvän dokumentaation piirteisiin.

Asiasanat:
dokumentointi, pelisuunnittelu, peliala

ALKULAUSE

Haluan kiittää BelleGames Oy -yrityksen johtajaa Pirjo Ritokangas-Huttusta työmahdollisuudesta sekä ohjaajaa Tuomo Tikkasta ja viestinnän opettajaa Tuula Hopeavuorta tuesta ja avusta. Haluan myös erityisesti kiittää työpariani Veli-Matti Jaakolaa hyvästä yhteistyöstämme projektin teon aikana.

Oulussa 5.2.2014

Tanja Henttunen

SISÄLLYS

TIIVISTELMÄ	3
ALKULAUSE	4
SISÄLLYS	5
1 JOHDANTO	6
2 DOKUMENTOINNIN TARKOITUS	7
3 DOKUMENTAATIOT PELISUUNNITTELUSSA	8
3.1 Visiodokumentti	9
3.2 Pelisuunnitelma	12
3.3 Tyyliohje	19
3.4 Tekninen suunnitelma	21
3.5 Testaussuunnitelma	23
3.6 Aikataulusuunnitelma	24
3.7 Riskianalyysi	25
3.8 Muut dokumentit	27
3.8.1 Palaverimuistio	27
3.8.2 Työpäiväkirja	28
3.8.3 Ohjelmakoodin kommentointi	28
3.8.4 Ylläpito-ohjeet	28
4 DOKUMENTOINNIN HYÖDYT JA HAITAT	30
4.1 Hyötyvaikutukset	30
4.2 Haittavaikutukset	31
5 HYVÄN DOKUMENTOINNIN PIIRTEET	33
6 YHTEENVETO	35
LÄHTEET	36

1 JOHDANTO

Opinnäytetyön aiheena oli dokumentointi pelialan yrityksessä. Sen rinnalla projektin aikana kehitettiin peliprojektia nimeltä "Year of Eden", joka julkaistiin Facebook-sivustolla sekä toteutettiin projektiin dokumentaatioita ja tutkimuksia. Opinnäytetyön idea syntyi harjoittelutyökokemuksistani peliyrityksissä, joissa dokumentoiminen toteutettiin eri tavoin.

Peliala on viime vuosina lähtenyt Suomessa nousuun, mikä on saanut useat unelmoimaan omasta peliyrityksestä (Capri 2010). Yritysten johtajat näkevät dokumentoinnin aikaa vievänä ja kalliina tapana toteuttaa projekteja ja siksi siihen ei panosteta tarpeeksi (Rollings – Ernest 2003, 13).

Ennen yksi henkilö pystyi tekemään pelin ilman dokumentaatiota, mutta nykyään pelit ovat niin laajoja, että yhden pelin tekemiseen tarvitaan 20–50 tekijän työpanos. Työryhmään kuuluu eri alojen osajia, kuten ohjelmoijia, suunnittelijoita ja graafikoita. Dokumentoinnin puuttuminen käy näin isossa yrityksessä kohtalokkaaksi. (Tervakari 2011.)

Tämä opinnäytetyö toimii ohjenuorana pelialan yrityksille, jotka haluavat muodostaa perustan dokumentoinnille. Opinnäytetyössä käydään läpi, miksi dokumentointi on tärkeä osa ohjelmointiyrityksessä, mitä dokumentointitapoja yleisesti käytetään, milloin siitä on hyötyä, milloin haittaa ja millaisia piirteitä helposti luettavalta dokumentilta voidaan odottaa. Opinnäytetyössä ei keskitytä pelien suunnitteluun tarkasti.

2 DOKUMENTOINNIN TARKOITUS

Yksinkertaisimmillaan dokumentoinnin tarkoitus on tiedonvälitys. Se antaa yrityksen sisällä oleville osaajaryhmille, kuten ohjelmoijille, graafikoille ja markkinaosaajille, yhteisen keskustelukielen projekteista puhuttaessa. Samalla kommunikointi helpottuu yrityksen ulkopuolisten henkilöiden, kuten tilaajan kanssa. (Versiohallinta ja ohjelmiston dokumentointi. 2012.)

Dokumentointia voi ajatella myös muistiinpanovälineenä. Suunnittelun aikana syntyvien ideoiden määrä pakottaa kirjaamaan ne ylös, jolloin saadaan täydellinen kuva projektin toimivuudesta jo ennen sen toteutuksen aloittamista. (Versiohallinta ja ohjelmiston dokumentointi. 2012.) Se ei kuitenkaan poista palaverien ja muiden tapaamisien tarvetta, vaan jokaisen osallisen mielipiteen saaminen on tärkeä osa dokumentointia (Ryan 1999).

Dokumentoinnin avulla todetaan projektin kulku määrityksien, suunnitelmien ja toteutuksien kautta onnistuneeksi projektin päättymisen jälkeen (Versiohallinta ja ohjelmiston dokumentointi. 2012). Projektin jatkokehityksen kannalta on tärkeää tietää, kuinka aiempi versio projektista on onnistunut tavoitteessaan.

Yrityksen sisällä dokumentointi merkitsee eri asioita osaajaryhmille. Tuottajille dokumentit toimivat mainonnan välineenä, suunnittelijoille taas tuottajan visiona ja tietona pelin ominaisuuksista. Ohjelmoijille ja graafikoille dokumentit ovat tuotteen tekemisen käyttöohjeita ja tapa ilmaista oma osaamisensa alallaan. (Ryan 1999.)

3 DOKUMENTAATIOT PELISUUNNITTELUSSA

Yritysten välillä dokumenttien ulkoasut, sisällöt ja nimet vaihtelevat merkittävästi, koska dokumenteille ei ole kehitetty yhteistä standardia. Joissakin tapauksissa dokumenttien nimityksiä käytetään myös ristiin. (Tervakari 2011.) Yleensä käytettävät dokumentit riippuvat projektin aiheesta ja siitä, kenelle dokumentti on suunnattu. Esimerkiksi potentiaalisia sijoittajia varten on dokumentissa tärkeää kuvata tarkasti projektin budjetti ja aikataulu.

Yhteisten standardien puuttuessa on hyvin vaikea määrittää tarkalleen, mitä dokumentteja tulisi käyttää. Käytän tämän dokumentin aikana tuotantosuunnitelman (engl. production plan) rakennetta, johon kuuluu seitsemän erilaista pelikokonaisuutta täydentävää dokumentaatiota:

- visiodokumentti
- pelisuunnitelma
- tyyliohje
- tekninen suunnitelma
- aikataulusuunnitelma
- testausuunnitelma
- riskianalyysi (Bethke 2003, 101–103).

Tärkeimmät dokumentit tuotantosuunnitelmassa ovat visiodokumentti, pelisuunnitelma ja tekninen suunnitelma (Freeman 1997). Niissä käsitellään koko projektin ydin: mikä projekti on kyseessä ja miten kannattava se on, mitkä ominaisuudet muodostavat projektin ja miten ne toteutetaan.

Alaluvuissa esitellään lyhyesti projektin aikana tarvittavat tuotantosuunnitelman dokumentaatiot sekä muita tärkeitä dokumentaatioita. Käytän opinnäytetyössä Tony Mannisen Pelisuunnittelijan käsikirjassa määrittelemiä dokumenttinimityksiä.

3.1 Visiodokumentti

Visiodokumentti (engl. concept paper) on ensimmäinen dokumentti, joka tehdään idean keksimisen jälkeen. Sen avulla kootaan yhteen projektin pääideat ja luodaan keino projektin myymiselle joko muille yrityksen jäsenille tai potentiaaliselle rahoittajalle. Yleensä jo tässä vaiheessa voidaan selvittää, onko projektia kannattavaa toteuttaa. (Manninen 2007, 189.) Visiodokumenttia kutsutaan myös nimillä pitch document, prospectus, preliminary proposal, concept document tai pre-proposal.

Vuonna 2002 Irrational Games -yritys teki myöhemmin valtavan suosion saaneesta BioShock-pelistä visiodokumentin. Vuonna 2010 dokumentaatio julkistettiin yrityksen kotisivuilla. BioShock-pelin visiosuunnitelma nähdään kuinka projekti on muuttunut sen alkuvaiheista julkaisuhetkeen. Suunnitelmasta puuttuvat muun muassa julkaisuversiosta löytyvät tärkeät hahmot, kuten big daddyt, splicersit ja little sistersit (Ramsay 2010). Bioshock-pelin visiodokumentti on nähtävissä verkossa osoitteessa <http://irrationalgames.com/insider/from-the-vault-may/#>.

Vaikka BioShockin visiodokumentti on 20 sivun pituinen, jo yhden sivun mittaisella suunnitelmalla voi saada lukijan kiinnostuksen heräämään. Lyhyttä visiosuunnitelmaa puhutellaan yleensä nimellä synopsis (Manninen 2007, 189). Seuraavat otsikot kertovat oleellisen informaation toteutettavasta projektista:

- alustava nimi
- julkaisualusta
- genre
- tiivistelmä
- kohderyhmä
- pelimekaniikat
- ansaintamalli (Green 2011).

Alustava nimi

Nimi toimii mainosviestinä lukijalle. Sen avulla mielenkiinto joko innostaa jatkamaan lukemista tai tukahtuu olemattomiin. Liian kulunut tai monimutkainen nimi voi vähentää lukijan kiinnostusta tuotetta kohtaan. Vältettäviä nimiä voisivat esimerkiksi olla Soldier X: A Shooter tai Mega Buster 5000. (Green 2011.)

Julkaisualusta

Pelien pelaamiseen tarvitaan alusta (engl. video game platform), jossa sitä voidaan suorittaa. Pelialusta voi vaihdella tietokoneista konsoleihin ja mobiililaitteisiin, kuten älypuhelimiin. Sen valinta vaikuttaa paljon pelin pelattavuuteen ja ominaisuuksiin. (Green 2011.)

Julkaisualustaa päättäessä pitää ottaa huomioon laitteen sisäisten komponenttien, kuten prosessorin ja muistin, asettamat rajoitukset (Green 2011). Mobiililaitteissa on sisäisten komponenttien lisäksi otettava huomioon näytön koko, joka on paljon pienempi kuin esimerkiksi tietokone tai pelikonsoli alustoilla.

Genre

Genreä käytetään tuotteen kategorioimisessa. Pelejä yleensä kategorioidaan pelimekaniikkojen eikä visuaalisuuden tai kerronnan mukaan (Apperley 2006, 6–7). Genre voi esimerkiksi olla seikkailu, strategia tai kauhu. Jos projekti ei kuulu mihinkään genrekategoriaan, mielikuvan saaminen pelistä voi olla vaikeaa.

Tiivistelmä

Tiivistelmän tarkoituksena on saada lukijan kiinnostus heräämään projektiin. Huomion herättämisen voisi esimerkiksi toteuttaa kuvaamalla tunteita herättävästi tarinan pääelementit tai pelin ominaisuuksia 2–3 lauseella. (Green 2011.)

Kohderyhmä

Kohderyhmä eli asiakaskunta liittyy tiiviisti genreen ja alustaan (Spasov 2013). Sitä tarkentavia kysymyksiä ovat muun muassa: millainen asiakas on, miksi hän on tuotteen asiakas, millaisista vastaavista tuotteista hän pitää ja mistä ominaisuuksista hän pitää tuotteessa (Manninen 2007, 143). Asiakkaan identiteetin

määrittämisessä voi ottaa huomioon potentiaalisen asiakkaan iän, sukupuolen, harrastukset, kansallisuuden ja peliosaamistaidon.

Varhaisessa vaiheessa valittu kohderyhmä auttaa suunnittelijoita eläytymään heidän ajatusmaailmaan ja siten luomaan voimakkaampia elämyksiä pelin kohderyhmälle (Manninen 2007, 143). Mitä rajatumpi kohderyhmä on, sitä helpompaa on myös pelin mainostaminen. Esimerkiksi tiivistelmätarinan kirjoittaminen onnistuu vaivattomammin, jos kohdekäyttäjä on noin 18-vuotias, päivittäin kolme tuntia pelejä pelaava mies eikä pelkästään mieshenkilö.

Pelimekaniikat

Pelimekaniikka kuvaa, mikä tekee tuotteesta erityisen. Joko siinä on uudenlainen toiminto tai hyödynnetään jo keksittyä mekaniikkaa uudella tavalla. Kuuden ominaisuuden lista antaa riittävän mielikuvan laajan projektin jatkotyöstämisen mahdollisuuksista. (Green 2011.) Prototyypin eli demoversion tuottamista var-ten pelimekaniikoista tulisi rajata ainoastaan tärkeimmät toteutettavaksi.

Pelimekaniikan mielikuvan tuottamiseen voidaan mallina käyttää muiden aiemmin koettujen pelien ominaisuuksia. Esimerkiksi Titanfall-peli eroaa muista räiskintäpeleistä antamalla pelaajan ohjata jättiläisrobotteja jalkavaen rinnalla.

Ansaintamalli

Mobiili- ja sosiaalisen median sovellusten suosion kasvaessa ovat yritykset saaneet uusia keinoja rahan ansaitsemiseen eli monetisaatioon tuotteesta. Peli voi olla kertaostos tai sovellus on ilmainen, jossa näytetään mainoksia ja käyttäjä voi muun muassa ostaa pelin sisäisiä, in-app-ostoksia saadakseen hyötyominaisuuksia käyttöön. (Hiltunen ym. 2013, 31.)

Visiodokumenttiin riittää pelkkä ansaintamalliratkaisun mainitseminen ilman, että esitetään syytä valitsemalle ratkaisulle. Pelisuunnitelmassa syvennyttään enemmän valittuihin ratkaisuihin.

3.2 Pelisuunnitelma

Pelisuunnitelma (engl. game design document eli GDD) on suunnitelmadokumentti, joka määrittelee projektin ominaisuudet eli työryhmän yhteisen visioon millainen lopputulos tulisi olemaan (McGuire – Jenkins 2009). Toisin sanoen se on laajempi versio visiodokumentista.

On tärkeää kertoa lukijoille päätettyjen peliominaisuuksien toimivuudesta, sillä näin lukija ymmärtää, mitä hyötyä hän saa tehdyistä päätöksistä. Projektin vaatimukset todennäköisesti muuttuvat myöhemmin, mutta pelisuunnitelman avulla projektin peruspiirteet ovat pysyviä. (Hackett 2007.)

Kuten muidenkin dokumenttityyppien, myös pelisuunnitelman sisältö vaihtelee sen mukaan, millaisesta projektista on kyse (McGuire – Jenkins 2009.) Jos projekti on laaja, osa dokumentin sisällöstä voidaan esittää erillisissä dokumenteissa. Esimerkiksi laajoissa tarinapohjaisissa peleissä on järkevämpää esittää juoni käsikirjoitusdokumentissa kuin osana pelisuunnitelmaa.

Pelisuunnitelma voi esimerkiksi sisältää seuraavat otsikot:

- sisällysluettelo
- tiivistelmä
- pelimekaniikat
- julkaisualusta
- tekoäly
- käyttöliittymä
- tyyliohje
- äänimaailma
- juoni, maailma ja hahmot
- kenttäsuunnittelu
- budjetti
- ansaintamalli
- riskit (McGuire – Jenkins 2009; Manninen 2007,138; Maxwell Chandler 2013, 167).

Joskus pelisuunnitelma yhdistetään teknisen suunnitelman kanssa. Tällöin dokumentissa on sekä pelin ominaisuuksia että kaavioita, joiden avulla ohjelmoijat toteuttavat projektin. Kaaviot voivat auttaa dokumentin visuaalisessa näytävyydessä ja luettavuudessa, mutta niiden ei pidä olla dokumentin pääosuus. Jotta päätösten motivaatio avautuisi lukijoillekin, on niistä kirjoitettava ymmärrettävällä tavalla. (Hackett 2007.)

Vuonna 1995 valmistui Race'n'Chase-pelin pelisuunnitelma. Tässä vaiheessa ei kukaan osannut arvata, että peli tulisi vuonna 1997 tuntemaan suuren suosion saaneena pelinä nimeltä Grand Theft Auto. Dokumenttiin on yhdistetty tyy-lisuunnan ja teknisen suunnitelman ominaisuuksia, mutta pääpiirteet ovat samat. GTA-pelin pelisuunnitelma löytyy osoitteesta

<http://www.scribd.com/doc/53563149/Grand-Theft-Auto-Design-Document>.

Sisällysluettelo

Kun tekstin pituus kasvaa yli 50 sivun, tiedonhakua varten tehty navigointijärjestelmä eli sisällysluettelo on olennainen osa dokumentaatiota. Kuitenkin se usein jää kiireen keskellä tekemättä vaikeuttaen tarvittavan tiedon löytämistä. (Rouse 2005, 360.)

Tiivistelmä

Projektin idean esittäminen ei ole välttämätöntä pelisuunnitelmassa, sillä kokonaisuus on jo esitelty visiodokumentissa. Uusille työntekijöille kuitenkin kokonais kuvan näkeminen voi olla helpompaa, jos he saavat lukea lyhyen version projektin ideasta (Rouse 2005, 360).

Pelimekaniikat

Pelimekaniikkojen määrittäminen on vaikein osa dokumentin suunnittelusta, koska ne ovat keskeisin osa pelikokonaisuudesta (Rouse 2005, 361–362). Visiodokumenttiin listatut toiminnalliset ominaisuudet eli pelimekaniikat tulisi esittää mahdollisimman yksityiskohtaisesti ilman, että teksti rönsyilee.

Tärkeitä huomioon otettavia asioita ovat, kuinka halutut mekaniikat toimivat keskenään ja miten monipuolisia ne ovat (Rouse 2005, 361–362). Visiosuunnitelman pelimekaniikat alakohdassa kuvattiin Titanfall-pelin eroamista muista

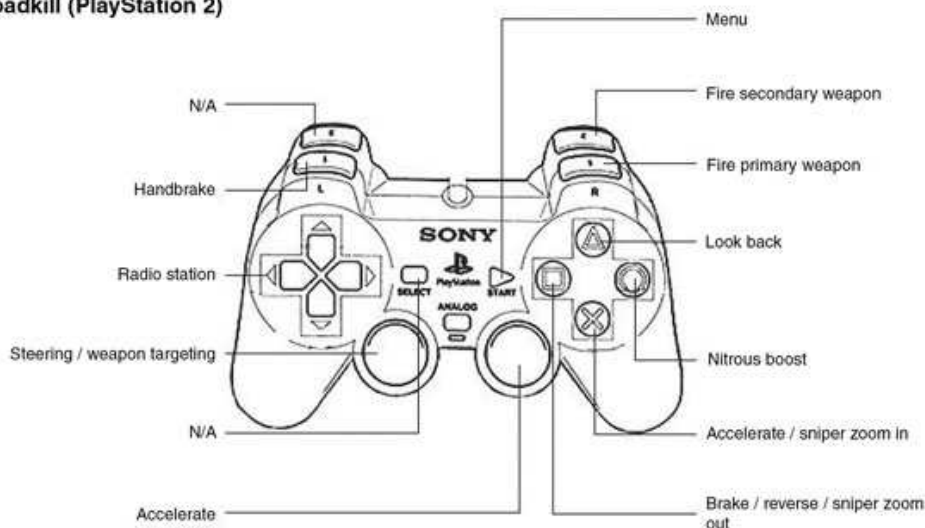
räiskintäpeleistä antamalla pelaajan ohjata jättiläisrobotteja jalkaväen rinnalla. Tämän pelimekaniikan monipuolisuutta ja toimivuutta voisi pohtia, riittäisikö ominaisuus pelkästään saamaan pelille ostajia ja antaako robotin käyttö sitä käyttäville liikaa ylivoimaisuutta muihin nähden.

Julkaisualusta

Pelin pelaamiseen tarvitaan julkaisualusta (engl. publishing platform), jossa sitä voidaan suorittaa. Alusta voi vaihdella tietokoneista konsoleihin ja mobiililaitteisiin, kuten älypuheliin. Sen valinta vaikuttaa paljon pelin pelattavuuteen ja ominaisuuksiin. Julkaisualustaa pohtiessa voi käyttää SWOT-analyysia, jossa käydään läpi vahvuudet, heikkoudet, mahdollisuudet ja uhat. (Green 2011.) Esimerkiksi projektin suuruus ja näytön koko olisivat tärkeitä pohdittavia aiheita, jos alustana olisi älypuhelin.

Julkaisualustaan kuuluu myös siinä käytettävä ohjaustapa. Pelit ovat riippuvaisia toimivasta ohjattavuudesta, sillä muuten hyvästä peli-ideasta tulee pelaa-miskelvoton (McGuire – Jenkins 2009). Esimerkiksi konsolipeleissä tottumukse-na on, että oikealla liipaisimella tapahtuu aseiden ampuminen, jos on kyse räis-kintäpeleistä (kuva 1).

Roadkill (PlayStation 2)



KUVA 1. Roadkill-pelin ohjattavuus Playstation 2 -ohjaimella (McGuire – Jenkins 2009)

Tekoäly

Tekoälyn tarkoitus on asettaa ei-elävä olento ajattelemaan, käyttäytymään ja toimimaan ihmismäisellä tavalla (ITL Education Solutions Limited 2011, luku 24.2: artificial intelligence: definition). Peleissä tekoäly näkyy esimerkiksi vastustajana tai keskusteltavana hahmona.

Tärkeä käsiteltävä osuus on, miten älykäs tekoälyn tulisi olla. Jos pelissä tulee esimerkiksi olemaan interaktiivisia hahmoja, suunnitellaan millä tavalla ne reagoivat pelaajan toimintoihin. Esimerkiksi kun pelaaja astuu tekoälyhahmon tielle, väistääkö hahmo pelaajaa vai tapahtuuko törmäys.

Käyttöliittymä

Käyttöliittymällä (engl. user interface) kuvataan elektronisen laitteen käyttämistä. Käyttöliittymästä käytetään yleisesti lyhennettä UI, johon kuuluvat muun muassa valikot, kuvakkeet, näppäimistön pikanäppäimet, hiiri ja gesturet eli sormen liiketoiminnot. (McGuire – Jenkins 2009.)

Pelit koostuvat erilaisista graafisista käyttöliittymänäkymistä eli GUI:sta. Osa näkymistä on valikkoja ja pelin aikana näkyviä HUD-näkymiä (head-up display) eli visuaalista informaatiota (Wilson 2006). Visuaalista informaatiota ovat Deus Ex -pelissä esimerkiksi energiamäärä ja aseet (kuva 2).



KUVA 2. Deus Ex -pelin käyttöliittymänäkymä (Wilson 2006)

Käyttöliittymää kuvataan yleensä luonnospiirroksin, havainnollistavin kuvin tai esimerkiksi käyttöliittymämallien avulla, jotta lukija saa selkeän käsityksen visuaalisesta ilmeestä, sijoittelusta ja muista osa-alueista. Käyttöliittymän kuvaukseen tarkentavia kysymyksiä voisivat olla, miten käyttöliittymä toimii ja miltä se näyttää, miten pelin suunniteltu perspektiivi tulee ottaa huomioon, miten paljon pelaaja voi kontrolloida käyttöliittymää ja minkälaisia mekanismeja käyttöliittymään tarvitaan. (Manninen 2007, 153–154.)

Tyyliohje

Tyyliohje on yhdistelmä kuvaavaa tekstiä ja hahmotelmakuvia, joilla pyritään kuvaamaan alustava visuaalinen ulkonäkö projektille (McGuire – Jenkins 2009.) Visuaalinen ulkonäkö on erittäin merkittävä peleissä ja sen vuoksi laajemmissa kokonaisuuksissa ulkoasu suunnitellaan erillisessä tyyliohje-dokumentissa. Dokumentin sisältöä esitellään luvussa 3.3.

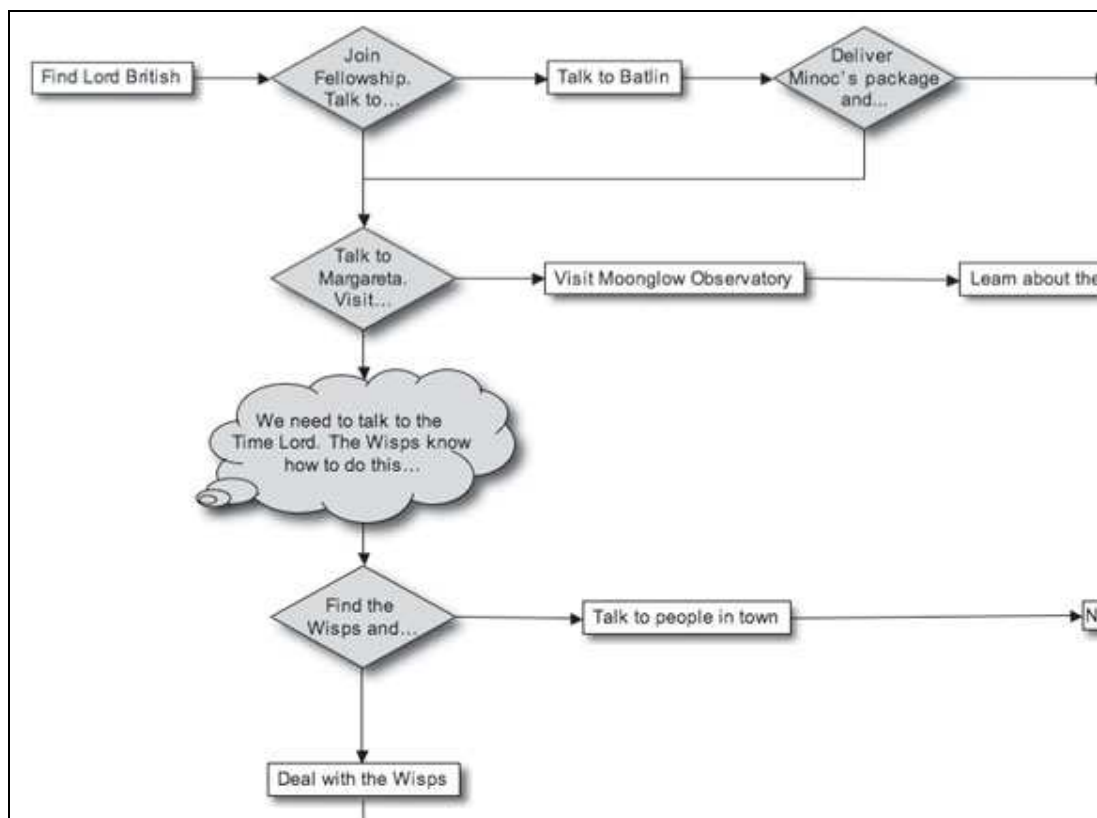
Äänimaailma

Äänimaailmaan kuuluvat musiikin lisäksi ääniefektit. Ääniefekti voi esimerkiksi olla painikkeen painamisesta kuuluva ääni, pelin tason läpäisemisessä kuuluva efekti tai aseiden laukaisun ääni.

Äänimaailman suunnitteluun voisi kuulua tyylikuvaus siitä, minkä tyyppistä musiikkia tai ääniefektejä projektissa käytetään. Myöhemmin on mahdollista kirjata projektissa käytetyt musiikkikappaleiden ja ääniefektien nimet sekä käyttötilanteet muistiin. Tämä helpottaa äänimaailman muokkausta ja suunnittelua.

Juoni, maailma ja hahmot

Peliprojekti, kuten esimerkiksi shakki, ei aina sisällä tarinaa, mutta joskus juoni on merkittävässä osassa koko pelissä. Juonenkulkuun vaikuttavat dialogi, tapahtumat, henkilöt ja tapahtumapaikka. Jos yksikin näistä osista on puutteellisesti suunniteltu ja toteutettu, vaikuttaa se pelikokonaisuuteen haitallisesti. Juoni voidaan suunnitella juonikaavioiden avulla (kuva 3). Kaavio helpottaa tarinan rytmittämistä ja kokonaiskuvan hahmottamista.



KUVA 3. Juonikaavion osa *Ultima VII* -pelistä (McGuire – Jenkins 2009)

Pelissä kuvattavan maailman määrittelemisen tarkkuus riippuu siitä, miten paljon pelaajalle annetaan vapautta tutkia ympäristöä. Halo-peleissä maailmakuvauksesta on olemassa satojen sivujen dokumentaatiot, jossa kuvataan hahmojen ja avaruusoliorotujen lisäksi tapahtumapaikkaa. Dokumentaatioon voi riittää alustavasti myös vain pelimaailman pohjan kirjaaminen. Pohja voi esimerkiksi olla päähahmon ympäristön kuvaus. (McGuire – Jenkins 2009.)

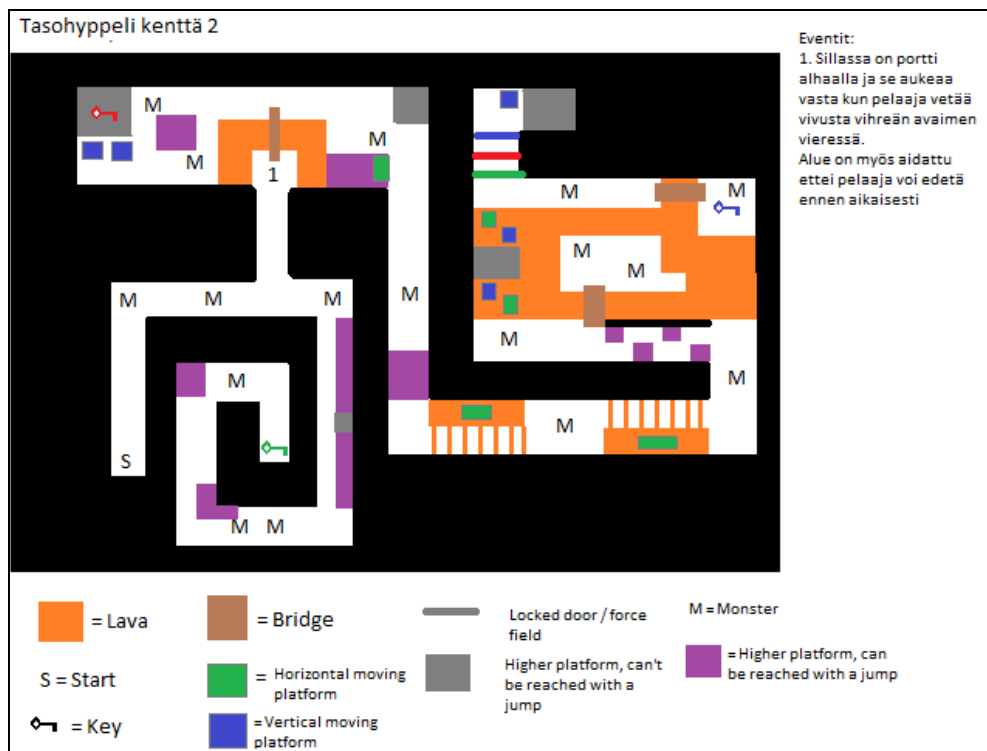
Pelinhahmoille tulee määrittää niiden taustat ja motivaatiot. Erityisesti tarinapainotteisissa peleissä päähenkilön tekemät ratkaisut kuvautuvat hänen motivaation kautta. Motivaation lisäksi muun muassa vaatteet, tausta, ystävät, ammatti ja mielenkiinnonkohteet vaikuttavat pelaajan näkemykseen henkilöistä. (McGuire – Jenkins 2009.)

Kenttäsuunnittelu

Lähes kaikki pelit ovat jakautuneita kenttiin eli tasoihin, jotka kuvaavat hienovaraisia eroavaisuuksia pelin vaikeustasoissa. Esimerkiksi strategiapeleissä eri vaiheet, kuten resurssien kerääminen, rakentaminen ja taisteleva voisivat

kaikki olla omia kenttiään, kun taas tarinapohjaisissa seikkailupeleissä kentät ovat osa suurempaa maailmaa, jota pelaaja tutkii edetessään juonen mukaisesti. (McGuire – Jenkins 2009.)

Kenttäsuunnittelun (engl. level design) avulla yksinkertaisimmillaan tutkitaan, millä tavalla projektin kentät ovat liitoksissa toisiinsa ja kuvataan, millaisia ne tulisivat olemaan (kuva 4). Liitoksilla tarkoitetaan tapaa, kuinka kenttien välillä liikutaan. (Manninen 2007, 150) Esimerkiksi useissa älypuhelinpeleissä kentän voi valita valikon kautta tai läpäisemällä aiemman. Kentän suunnitteleminen on todella laaja prosessi, sillä se vaikuttaa koko projektin pelattavuuteen.



KUVA 4. Kenttäsuunnitelma harjoitteluprojektille

Budjetti

Budjetti kuvaa projektituotannon arvioidut kulut. Siihen kuuluvat muun muassa henkilökulut, laitteistot, ohjelmistot ja ulkopuoliset graafiset tai musiikilliset resurssit. Budjetin käsitteleminen isoissa projekteissa on monimutkaista ja siksi sen käsittelemisessä on yleensä oma vastuhenkilönsä ja dokumentaationsa. (McGuire – Jenkins 2009.)

Ansaintamalli

Ansaintamalli eli monetisaatio tarkoittaa keinoja rahan saamiseksi tuotteen avulla. Sen tärkeys on mobiililaitteiden ja internetin ilmaispelien (engl. free to play) avulla noussut yhdeksi tärkeimmistä osista projektin suunnittelussa. (Hiltunen ym. 2013, 31.)

Visiodokumentissa esitelty ansaintamalli tulee käsitellä perusteellisesti pelisuunnitelmassa. Jos on tarve hyödyntää pelien sisäisiä in-app-ostoksia, tulee peliprojektista löytää jokin ominaisuus, jonka käyttäjä voisi haluta ostaa. Yksinkertaisimmillaan peleissä, joissa pelaaja voi luoda oman hahmon, in-app-ostos voisi olla pelihahmolle uusi ulkoasu.

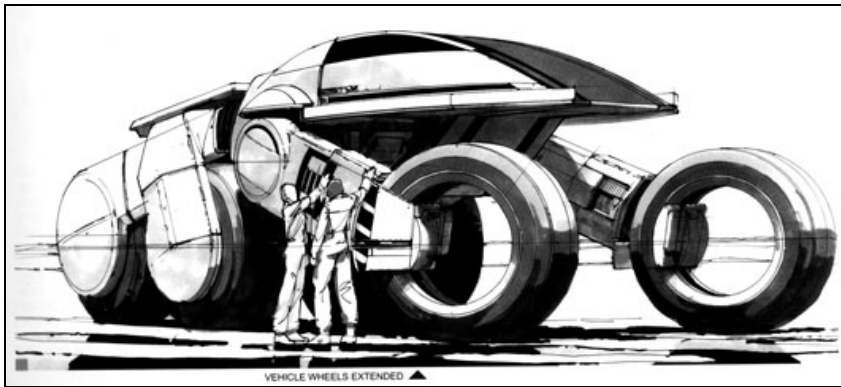
Toimivan ansaintamallin kehittämiseen käytetään paljon aikaa sekä resursseja ja joskus siitä on vastuussa oma henkilöstönsä. Tämä johtuu siitä, että ilmaisissa mobiili- ja internetsovelluksissa kaikki yrityksen tuotot ovat riippuvaisia ansaintamallin toimivuudesta.

Riskit

Mahdollisten ongelmien eli riskien määrittäminen auttaa minimoimaan niiden vahingon suuruutta (Spasov 2013). Pienemmissä projekteissa riskimäärittäminen voidaan tehdä pelisuunnitelmassa, mutta riskeistä on oma hallintadokumentaationsa, joka on esitetty luvussa 3.7.

3.3 Tyyliohje

Tyyliohje (engl. style guide tai art bible) määrittelee pelin tyyliuunnan eli esi-
neiden, ympäristön ja hahmojen ulkonäön. Tyyliä kuvataan konseptikuvien (kuva 5), väripalettien ja muiden visuaalisia esimerkkien avulla. (Maxwell Chandler 2013, 168.) Kuvamateriaalin rinnalla voi myös olla sitä tukevaa ja selittävää tekstiä, jonka avulla kuvataan esimerkiksi erityispiirteitä tai kriittisiä alueita. Päägraafikko on yleensä vastuussa dokumentaation teosta, vaikka pelisuunnittelija tekeekin tiivistä työtä hänen kanssaan graafista ilmettä päätettäessä. (Manninen 2007, 190.)



KUVA 5. Syd Mead, konseptikuvataiteen isoisä kuvaa industrialistista kulkuneuvoa harmaasävyin (Pickthall 2012)

Dokumenttiin voi tarvittaessa lisätä resurssinlistan ja työkalujen käyttöohjeet. (Maxwell Chandler 2013, 168). Hyvä esimerkki on The Trip -pelin tyyliohje, jonka voi ladata osoitteesta <http://www.roguearbiter.com/port/theTrip/The%20Trip%20-%20Art%20Bible.pdf>. Kuvamateriaalin lisäksi dokumentti sisältää kuvaa-vaa tekstiä pelin maailmasta ja kentistä. Käyttöliittymän konseptikuva on myös lisätty dokumenttiin.

Resurssilista

Resurssilistaan (engl. asset list) kuuluvat kaikki taiteelliset ominaisuudet, jotka projektiin tarvitaan. Resursseja ovat esimerkiksi hahmomallit, tasot, videot, tekstuurit ja muut visuaaliset elementit. Resurssien priorisoiminen eli listaaminen tärkeysjärjestykseen on erityisen olennaista, jotta projektin pääresurssit tulisivat varmasti valmiiksi aikataulun mukaisesti. (Maxwell Chandler 2013, 168.)

Työkalujen käyttöohjeet

Työkalujen käyttöohjeilla (engl. tool instructions) tarkoitetaan ohjeistusta projektin aikana käytettäviin grafiikkaohjelmistoihin, kuten tasonluonti- tai valaistustyökaluun (Maxwell Chandler 2013, 168). Ohjeiden tekeminen ei ole pakollista, mutta se voi auttaa perehdytyksessä jos projektiryhmään tulee uusia työntekijöitä.

3.4 Tekninen suunnitelma

Tekninen suunnitelma (engl. technical design document) on toteutussuunnitelma projektin ominaisuuksille. Dokumentin tarkoitus on saada projektin halutut ominaisuudet muotoon, josta pystytään luomaan toteutusvaiheessa tarvittavia kaavioita ohjelmoijien käyttöön. (Bethke 2003, 136.) Sen tekeminen on pääohjelmoijan vastuulla, sillä hänen vastuunaan on usein myös projektin toteutuksen valvonta (Rouse 2005, 317).

Tekniseen suunnitelmaan kuuluvat muun muassa vaatimusmäärittely, ohjelmointikäytännöt, luokkahierarkiat ja testaussuunnitelma (Sikora 2002; Bethke 2003, 136). Siitä, kuten kaikista muistakin dokumenteista, on olemassa loputtomasti erilaisia versioita, joissa panostetaan projektin tarkoituksen mukaan eri aihealueisiin. Usein tekninen suunnitelma nähdään yhdistettynä pelisuunnitelman osaksi.

On tärkeää huomioda, että tekninen suunnitelma on tarkoitettu ainoastaan ohjelmointiin tarkoitettuja tietoja varten. Siihen ei tule lisätä pelisuunnitelman tietoja, kuten kuvainnollista tietoa tarinasta, hahmoista tai markkinoinnista. Dokumenttiin ei kuulu myöskään lisätä suoraa ohjelmakoodia, sillä ohjelmakoodit vaikeuttavat dokumentin päivittämistä. (Sikora 2002.)

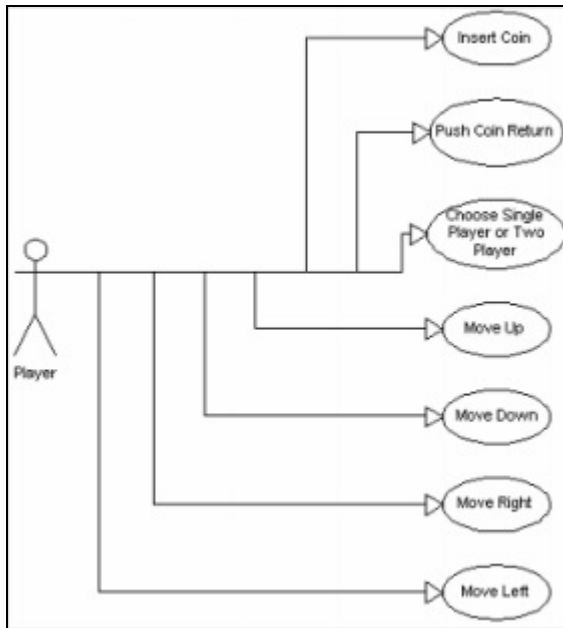
Vaatimusmäärittely

Vaatus tarkoittaa projektin haluttua ominaisuutta, joka projektiin tulee toteuttaa. Se ei pelkästään ole mekaaninen osa lopputuloksessa, vaan mukana ovat myös esimerkiksi ääniin ja graafisiin ominaisuuksiin vaikuttavat ominaisuudet.

Projektin haluttuja vaatimuksia voi määrittää useilla eri tavoilla. Ketterän suunnitelmamallin mukaisesti vaatimukset luodaan käyttäjätarinoiden avulla. Käyttäjätarinoissa vaatimus kerrotaan pelaajan näkökulmasta, esimerkiksi: ”Käyttäjänä haluan, että päästessäni kentän läpi saan arvosanan suoriutumisestani”.

Toinen tapa vaatimusten määrittämiseen on käyttää requirements capture-mallia, jossa luodaan ominaisuuksista käyttötapauskaavioita (engl. use case diagram). Kuvassa 6 näkyvässä Pac-Man-pelissä käytetään pelaajan tekemiä

syöttötoimintoja (engl. player input) kuvaamaan, mitä toimintoja hän voi tehdä. (Bethke 2003, 82–83.)



KUVA 6. Use case -malli pelaajan syöttötapauksia Pac-Man-pelissä (Bethke 2003, 85)

Vaatimuksissa on otettava huomioon järjestelmävaatimukset laitteesta, jossa peliä suoritetaan (Sikora 2002). Järjestelmävaatimuksia voisivat olla alustan muistin määrä ja näytön resoluutio. Mobiili- ja tietokonealustoilla järjestelmävaatimukset vaihtelevat laitteittain, joten minimivaatimusten määrittäminen on olennaista.

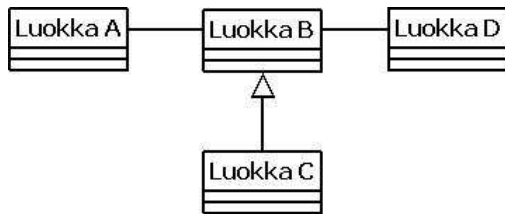
Ohjelmointikäytännöt

Ohjelmointikäytännöllä tarkoitetaan sääntöjä, joiden mukaan projekti toteutetaan. Käytäntönä voisi esimerkiksi olla ohjelmoinnissa käytettävien funktioiden ja muuttujien nimeämistavat ja järjestykset ohjelmakoodissa (Sikora 2002). Säännöt ovat erityisen tärkeitä useamman henkilön projektiryhmässä, joka ei ole aiemmin tehnyt yhteistyötä keskenään.

Luokkahierarkiat

Jotta ohjelmitava kokonaisuus saataisiin mahdollisimman helposti luettavaan muotoon, tehdään sen rakenteesta kaavioita (kuva 7). Kaaviot selkeyttävät pro-

jektin haluttuja ominaisuuksia ja yhdenmukaistavat ohjelmoijien ohjelmointitapoja.



KUVA 7. Esimerkki luokkakaaviosta

Testaussuunnitelma

Jokaisen projektin loppuvaiheessa tulisi tapahtua testausta virhetilanteiden varalta. Testaussuunnitelman (engl. testing plan) voi toteuttaa myös erillisessä dokumentaatiossa, joka käsitellään luvussa 3.5. Erillisen dokumentaation tekeminen voi olla järkevää, jos testattavien ominaisuuksien määrä on suuri tai dokumentaatiota tarvitsee testauksen aikana useampi henkilö.

3.5 Testaussuunnitelma

Kun projektin ensimmäinen versio on valmistunut, se on kaukana täydellisyydestä. Mitä monimutkaisempi projekti on kyseessä, sitä todennäköisemmin siinä on virheitä, joita ei toteutusvaiheessa ole otettu huomioon. Siksi on tärkeää, että projektille on olemassa testausryhmä, joka pyrkii hakemaan valmiista tuotteesta vikoja eli suorittaa virhearviointia projektista.

Testaussuunnitelmassa (engl. testing plan) päätetään, kuinka projektin virhearviointi toteutetaan. Suunnitelman voi jakaa kahteen osaan: ensimmäisessä osassa määritetään testaustapa ja -työkalut, aikataulu ja vastuuhenkilöt ja toisessa osassa määritetään testattavat ominaisuudet. (Japenga.) Testaustapoja on useanlaisia kuten esimerkiksi testaajat voivat yrittää rikkoa peliä kaikin mahdollisin tavoin tai he voivat testata listattujen ominaisuuksien toimivuutta. Kolmas vaihtoehto on käyttää betatestausta, jossa rajattu ryhmä potentiaalisia asiakkaita pelaa viimeistelemätöntä pelin versiota, mitä tarkkaillaan ja dokumentoidaan. (Bethke 2003, 155.)

Testattavat ominaisuudet voidaan ottaa teknisen dokumentaation vaatimuksista (Bethke 2003, 155). Jos vaatimukset ovat puutteelliset tai niitä on liian vähän, voi projektista esittää kysymyksiä ja niiden avulla toteuttaa testitapauksia (Japenga). Kysymys voisi esimerkiksi olla, mitä pelikokonaisuuksia pelissä on, mikä on sen suoritusnopeus ja miten paljon ominaisuudet syövät laitteelta tehoja tai miten on tietoturvallisuus toteutettu.

3.6 Aikataulusuunnitelma

Koska pelialalla projektin tuottaminen on pitkäkestoista ja käytettävät budjetit ovat suuria, projektin aikataulun suunnittelu on erittäin tärkeää. Yleisesti aikatauluttamisella tarkoitetaan sen arvioimista, kuinka kauan projektin ominaisuuksien tuottamiseen tulisi maksimissaan käyttää aikaa. Aikataulusuunnitelman (engl. project schedule) avulla määritetään jokaisen työntekijän työpanos. Sitä kutsutaan myös nimellä project plan tai project planning.

Aikataulutuksen tunnetuimmat menetelmät ovat Gantt-kaavio, PERT-kaavio ja kriittisen polun menetelmä (engl. critical path method) (Bethke 2003, 157). Menetelmiä ei käsitellä tässä opinnäytetyössä.

Aikataulutuksen vaiheet ovat seuraavanlaiset:

- projektin osittaminen
- tehtävien luettelointi
- tehtävien riippuvuuksien selvittäminen
- ajoittaminen kalenteriin
- tarkistus, analysointi ja ylläpito (Varanki 2010).

Projektin osittaminen

Projektille tulee määrittää karkeasti pääprojektivaiheet, kuten suunnittelu, toteutus, testaus ja ylläpito, sekä niille toimivat aikataulut. Aikataulujen sisällössä ovat tärkeitä vaiheiden kestot sekä aloitus- ja lopetuspäivämäärät. Päivämäärissä tulee ottaa huomioon viikonloppujen, lomien ja muiden menojen määrät. Projektin osittamisen avulla myöhemmin tapahtuva tehtävien luettelointi on helpompaa, sillä projektilla on yleinen määrittäminen siitä, mitä vaiheita siinä on. (Varanki 2010.)

Tehtävien luettelointi

Tarkemman aikataulusuunnitelman avulla määritellään kaikki yksittäiset tehtävät, niiden kestot sekä aloitus- ja lopetuspäivämäärät (Varanki 2010). Yksittäinen tehtävä ei ole ainoastaan projektin tuottamisessa tehtävä osa, vaan mukaan kuuluvat myös esimerkiksi dokumentointi ja palaverit.

Suuret projektit on usein jaettava pienempiin osaprojekteihin, jotta tehtävien jako on toteutettavissa (Varanki 2010). Esimerkiksi GTA V -pelissä jokaiselle minipelille, kuten tennikselle ja golfille, ovat tekijät tehneet omat osaprojektinsa.

Tehtävien riippuvuuksien selvittäminen

Tehtävien riippuvuuksilla voidaan selvittää, onko mahdollista toteuttaa tehtäviä samanaikaisesti ja missä järjestyksessä kokonaisuudet tulee suorittaa (Varanki 2010). Esimerkiksi painikkeen toimintoa ei voida luoda, ennen kuin se on sijoitettu pelinäkymään.

Tarkistus, analysointi ja ylläpito

Aikataulua analysoidaan koko projektin ajan. Sen avulla tutkitaan, onko projektin tehtävien aikataulu pysynyt kohdallaan ja sen perusteella päätetään aikataulun jatkotoimet. On hyvä jättää jokaiseen työtehtävään työvaraa mahdollisten ongelmien ratkaisemista varten.

3.7 Riskianalyysi

Mitä isommasta projektista on kyse, sitä suurempi riski on ongelmien ilmenemiselle. Riski nimityksellä kuvataan projektissa mahdollisesti tapahtuvaa ongelmaa, kuten laite- tai organisaatio-ongelmaa, johon voidaan vaikuttaa valmistautumalla.

Riskianalyysinä tunnettu risk management plan tai risk mitigation plan on tärkeä IT-alan dokumentaatio, sillä sen avulla voidaan välttää ongelmatilanteet tai niiden sattuessa on olemassa varasuunnitelma.

Suunnitelma esitetään taulukkomaisesti (taulukko 1). Sen yksinkertaisimmassa versiossa kuvataan seuraavat otsakkeet: ID, kuvaus, todennäköisyys, vahinko-

määrä, vahinkoarvo, prioriteetti ja lievennyskeinot (Chemuturi 2013, luku 5: Risk Management Plan).

TAULUKKO 1. Yksinkertaisen hallintataulukon pohja (Chemuturi 2013, luku 5: Overall Project Schedule)

ID	Kuvaus	Todennäköisyys	Vahinkomäärä (\$)	Arvo (\$)	Prioriteetti	Lievennyskeino

Riskianalyysiin voi yhdistää pelisuunnitelmaan tai tekniseen suunnitelmaan, mutta riskitapausten päivittäminen voi olla helpompaa, jos niiden määritykset on tehty erilliseen dokumentaatioon.

ID

ID:llä tarkennetaan yksittäisiä riskitapauksia ilman, että tarvitaan jokaiselle riskille tunnistettavaa nimeä (Chemuturi 2013, luku 5: Risk Management Plan). Se poistaa epäselvyydet samantyyppisistä riskikuvauksista. Riskin ID arvot tulevat juoksevan numeroinnin avulla eli uusimman riskitapausten id on riskitapausten senhetkinen määrä.

Kuvaus

Kuvauksella todennetaan mahdollinen riskitapaus (Chemuturi 2013, luku 5: Risk Management Plan). Yleensä kuvauksesta pituus on pisimmillään yhden lauseen mittainen, jotta se ei olisi liian vaikeasti ymmärrettävissä. Riskikuvaus voi olla esimerkiksi laitevika tai rahoituksen loppuminen.

Todennäköisyys, vahinkomäärä ja -arvo

Jokaiselle riskimahdollisuudelle tulee arvioida sen tapahtumisen todennäköisyys ja vahinkomäärä rahayksikkömuodossa. Vahingon arvon saa kertomalla todennäköisyysarvon lasketulla vahinkomäärällä. (Chemuturi 2013, luku 5: Risk Management Plan.)

Esimerkiksi riskin todennäköisyys on 50 % ja vahinkomäärä 1000 €, joten vahingon arvo on $0,5 * 1000 \text{ €} = 500 \text{ €}$.

Prioriteetti

Mitä tärkeämpi riski on kyseessä, sitä suurempi prioriteettiarvo sille tulisi asettaa (Chemuturi 2013, luku 5: Risk Management Plan). Esimerkiksi projektintilaa järkyttävän riskin tulisi olla prioriteettitaulukon kärjessä, kun taas projektia hidastavat tai todella epätodennäköiset riskit ovat alemmalla tasolla prioriteettitaulukossa.

Yleisimmin käytössä ovat luokittelu, jossa arvo voi olla 1–5, jossa viisi tarkoittaa suurta ja yksi pientä riskiä. Jos kaksi prioriteettiarvoa on samansuuruisia, suuremman todennäköisyyden saanut prioriteetti on korkea-arvoisempi (Chemuturi 2013, luku 5: Risk Management Plan).

Lievennyskeino

Kaikkien riskien tapahtumista ei voi estää, mutta pohtimalla etukäteen toimintasuunnitelman voi lieventää riskin haitallisuutta (Chemuturi 2013, luku 5: Risk Management Plan). Jos mahdollinen riski olisi laitteiston rikkoutuminen, olisi riskin lievennys esimerkiksi projektin versiohallinnan sijoittaminen ulkoiselle palvelimelle.

3.8 Muut dokumentit

Tuotantosuunnitelman ulkopuolelle jää tärkeitä dokumentteja, joista useat tehdään vasta projektin tekemisen aikana. Dokumentit on tarkoitettu helpottamaan asioiden muistamista ja kehittämään työntekijöiden osaamista jatkokehitystä varten.

3.8.1 Palaverimuistio

Jotta palaverissa päätetyt ja sovitut asiat olisivat todistettavissa, niistä tehdään oma muistio. Yleensä palaverin alussa päätetään sihteeri, joka tekee muistiinpanoja palaverin aikana. Palaverin jälkeen hän tekee muistion, joka jaetaan palaverin jäsenille. Muistion tekemiseen ei kuitenkaan tule käyttää kohtuuttomasti aikaa, sillä yleensä palaverin jäsenille riittää pelkkä tiivis listaus päätetyistä asioista. (Tervakari 2008.)

Palaverimuistiossa on yleensä seuraavanlaiset otsikot:

- projektin nimi, paikka ja päiväys
- osallistujat
- käsitellyt asiat
- päätetyt asiat
- jatkotoimenpiteet eli seuraavan palaverin päivämäärä ja tehtävät työtehtävät. (Tervakari 2008.)

3.8.2 Työpäiväkirja

Työpäiväkirja auttaa työntekijää työnteon seuraamisessa ja kommentoimisessa sekä taitotason kehityksessä. Siihen merkitään päivän aikana tehdyt asiat, ajankäytön onnistuminen, esiintyneet ongelmat ja löydetty ratkaisukeinot. (Tervakari 2008.) Työpäiväkirja auttaa myös työnantajaa työntekijän työpanoksen seuraamisessa.

3.8.3 Ohjelmakoodin kommentointi

Kommentointi on tärkeä osa dokumentointia, jos kyseessä on jatkokehitettävä tai iso projekti. Kommentoiminen auttaa tekijää ja jatkokehittäjiä ymmärtämään koodin tekijän päätökset.

Kuitenkin tulee ottaa huomioon, että kaikki kommentointi ei ole hyväksi. Yleistietämyksellä ymmärrettävät asiat tai todella pitkät kommentit tulee jättää väliin, sillä kommentin ymmärtämisessä voi kulua sama aika kuin jos työntekijä itse tutkisi koodin läpi. Myös vanhaa tietoa sisältävät ja poistettavat kooditekstit ovat turhia. (Hirschlein 2010.)

3.8.4 Ylläpito-ohjeet

Projektin dokumentaatioiden ylläpitämistä eli iteroimista tulisi tapahtua koko projektin ajan. Jotta dokumentaatioita ei voi muokata mistä tahansa syystä, luodaan niiden päivittämiselle yhteiset säännöt. Yleensä dokumentaation ylläpitämiselle valitaan vastuuhenkilö, jotta sääntöjä noudatettaisiin varmasti.

Ylläpito-ohjeissa tulee ottaa huomioon, miten projektiin voidaan lisätä uusia ominaisuuksia ja mitä asioita otetaan huomioon muutoksen tehtäessä (Slander). Huomioitava asia voisi esimerkiksi olla dokumentaatiomuutoksen ilmoitus-tapa kaikille projektin jäsenille. Ylläpito-ohjeet voidaan joko luoda erilliseen dokumentaatioon tai dokumentin viimeiseen lukuun.

4 DOKUMENTOINNIN HYÖDYT JA HAITAT

Useissa yrityksissä dokumentoiminen nähdään turhana työvaiheena (Capri 2010). Tähän mielipiteeseen ovat voineet vaikuttaa ennakkoluuloisuus tai aiemmat huonot kokemukset. Vaikka dokumentointia pidetään tärkeänä osana projektia, joskus erinäiset ongelmat ja piittaamattomuus tekevät siitä turhan. Tässä luvussa käydään läpi dokumentoinnin hyviä sekä huonoja puolia.

4.1 Hyötyvaikutukset

Vaikka dokumentointi on kallista, hyvin tehtynä ja päivitettyinä se on kannattavaa. Hyvä dokumentaatiotaito antaa ulkopuolisille ammattimaisen kuvan yrityksestä ja pitää projektikokonaisuuden kasassa.

Suunnitelman realistisuus

Projektin voi ensi silmäyksellä vaikuttaa loistavalta liikeidealta, mutta lopputulos ei koskaan näytä samalta kuin alussa on määritetty. Dokumentoinnin avulla pakotetaan kirjaamaan ylös ajatustasolla olevat ideat ja määrittämään ovatko ne toteutuskelpoisia ajan ja resurssien puitteissa. (Ryan 1999.)

Yhteinen kommunikointitapa

Projektiryhmän sisäisillä jäsenillä, kuten graafikoilla ja ohjelmoijilla, sekä ulkoisilla jäsenillä, kuten sijoittajilla on omat osaamisalueensa. Dokumentoinnin avulla helpotetaan keskustelua projektin osallistujien kesken, sillä ne pyritään kirjoittamaan kielellä, jota kaikki pystyvät ymmärtämään. (Versiohallinta ja ohjelmiston dokumentointi. 2012.)

Projektin läpiviennin helpottaminen

Dokumentoinnin avulla varmistetaan, että projektissa toteutetaan ne ominaisuudet, jotka on suunniteltu (Spasov 2013). Se toimii kuin muistilista, johon ajatukset ja päätökset kirjataan yhteisten sopimusten muistamisen helpottamiseksi (Kuntz 2013).

Ajantasaisuus

Kun projektin tapahtumista on aikataulu, työntekijöillä on käsitys mihin aikaan mennessä ominaisuuksien tulisi olla valmiina ja mitä kunkin osapuolen tulisi tehdä (Spasov 2013). Aikatauluun tulee väistämättä poikkeuksia suuntaan tai toiseen, mutta perusrunko eli tapahtumajärjestys on hallinnassa.

Kun uusi työntekijä saapuu työpaikalle, hän saa perehdytyksestä enemmän irti, jos projektin tilanne ja tavoitteet on selvästi määriteltä (Kuntz 2013). Jos ohjelmointitavoista, kuten muuttujien nimeämistavoista, on erikseen sovittu, työntekijän ei itse tarvitse hakea tarvittavia tietoja, vaan hän pääsee heti työhön.

Kehitystyön helpottaminen

Kun koko projektin vaiheet ovat kirjattuna ylös, on jälkeensä ja projektin aikana mahdollista tehdä parannusehdotuksia tehtyihin päätöksiin. Koodin ylläpitäminen ja kommentointi auttavat myöhemmässä vaiheessa ohjelmakoodin uudelleen muokkaamista. (Spasov 2013.) Dokumentaatioiden avulla työryhmällä on enemmän kokemusta ohjelmakoodista ja sen rakenteesta.

4.2 Haittavaikutukset

Epäonnistunutta dokumentaatiota ei ole olemassa. Sen sijaan voidaan puhua tarpeettomasta dokumentoinnista eli ylidokumentoinnista, joka on tehty ilman suunnittelua. (Capri 2010.) Vaikka useissa tapauksissa enemmän on parempi, ylidokumentointi tuhlaa asiakkaan ja yrityksen rahoja ja siksi sitä tulee välttää. Luvussa käymme läpi viisi esimerkkitapahtumaa tarpeettomasta dokumentoinnista.

Lyhyt voimassaoloaika

Jos dokumentoitava asia on ajankohtainen vain lyhyen ajan, esimerkiksi kaksi viikkoa, virallisen dokumentin määrittäminen aiheesta on ajan ja resurssien tuhlausta. Jos jokainen todennettu asia kirjattaisiin, kansiot täyttyisivät dokumenteista, jotka ovat jatkossa tarpeettomia. (Vuori 2010.)

Jälkidokumentointi

Jälkidokumentointi tarkoittaa muistiinpanoja, jotka tehdään kun projekti on siinä vaiheessa, ettei niitä enää tarvitse tehdä. Yleensä dokumentointi on tarkoitettu

apukeinoksi suunnittelijoille projektin työstämistä varten ja siksi ei ole tuottavaa tehdä dokumentaatiota asiasta, josta ei ole enää hyötyä. (Berger 2010.)

Akateemisesti vaikeat dokumentit

Suunnittelijat joutuvat usein käsittelemään dokumenteissa vaikeasti ymmärrettäviä asioita (Schubert 2007). Vaikka tekijä ajattelisi asian olevan helposti ymmärrettävissä, hyvin usein aiheesta tietämätön lukija ei saa dokumentista tarvittavaa tietoa. Ongelman voi korjata joko kirjoittamalla niin selkeää tekstiä, jota kuka tahansa voi asian ymmärtää, tai tekemällä dokumentin alkuun luvun sanelityksistä, jossa käydään vaikeasti ymmärrettävät määritelmät läpi.

Ulkoasultaan heikot dokumentit

Jos dokumentin sisältöön, ulkoasuun ja pituuteen ei panosteta tarpeeksi, tekstistä tulee liian vaikeasti luettava ja tärkeitä aihekokonaisuuksia joudutaan etsimään kauan (Capri 2010). Nämä tapaukset tekevät dokumentista lähes turhan. Dokumentin luettavuutta voi testata asettamalla ulkopuolisen henkilön lukemaan tekstin läpi ja kysymällä häneltä mielipiteitä.

Iteroimattomat dokumentit

Iteroimisella tarkoitetaan projektin dokumenttien uudelleen läpikäymistä. Sen avulla tutkitaan projektin senhetkinen tilanne, ovatko suunnitelmat samanlaiset kuin on alun perin suunniteltu. Dokumentteja iteroidaan koko projektin ajan tarkentaen niiden ongelmakohtia. (Sylvester 2013, luku 11. Planning and Iteration)

Jos iteroiminen on hidasta tai se jätetään kesken, dokumenteista jäävät jäljelle vain vanhat versiot (Schubert 2007). Vanhojen versioiden iteroiminen on sitä vaikeampaa ja kalliimpaa tehdä, mitä kauemman aikaa edellisestä kerrasta on.

5 HYVÄN DOKUMENTOINNIN PIIRTEET

Vaikka dokumentin sisältö on kirjoitettu hyvin, voi se silti vaikuttaa lukijan mielestä vaikeasti ymmärrettävältä. Suoraa hyvän dokumentaation mallia ei ole olemassa, mutta seuraavat kappaleet voivat auttaa dokumentaation valmistuksessa.

Kohderyhmään keskittyminen

Kohderyhmä vaikuttaa paljon siihen, kuinka dokumentti tulisi suunnitella. Suunnittelijaryhmä haluaa päästä yhteisymmärrykseen päätöksistä, kuten visuaalisesta näkemyksestä ja juonenkulusta. Ohjelmoijat haluavat tietää, miten projekti toteutetaan eli mitä työtehtäviä projekti tarjoaa. Ymmärrettävin dokumentointitapa olisi tässä tilanteessa luettelomerkein tehty dokumentaatio, jossa on listattu ainoastaan tärkeimmät asiat. Tuottajille tärkeintä ovat aikataulun päivämäärät ja keino saada rahallista tuottoa. Laaduntarkastajille testisuunnitelma on keskeisin dokumentti ja sijoittajat haluavat saada ratkaisun päätöksiinsä. (Schubert 2007.)

Visuaalinen monipuolisuus

Kuten luvussa Dokumentin hyödyt ja haitat mainittiin, dokumentin visuaalinen ulkonäkö vaikuttaa hyvin paljon luettavuuteen. Visuaaliseen ulkonäköön kuuluvat myös kuvat. Sanonta ”yksi kuva kertoo enemmän kuin tuhat sanaa” ilmaisee täydellisesti kuvien kannattavuuden (Schubert 2007). Sijoittajien on paljon helpompaa antaa mielipiteensä projektista, jos he ovat nähneet kuvan miltä lopputulos tulisi näyttämään. Jos projektille ei ole käytössä kuvamateriaalia, havainnollistavana kuvana voi esimerkiksi käyttää muiden vastaavien projektien kuvamateriaalia.

Jos kuva- tai videomateriaalia ei ole missään muodossa käytössä, esimerkkiteksteillä voidaan yhtälailla havainnollistaa vaikeaselkoisia tai monella tavalla ymmärrettäviä asioita (Schubert 2007). Dokumentaatiossa esimerkkiteksti voisi olla seuraavanlainen: ”Pelaajan päästessä tason läpi ruudulle ilmestyy yhdestä kolmeen tähteä sen mukaan, miten hyvin hän tasossa onnistuu.”

Helppo luettavuus

Jotta teksti olisi mahdollisimman helposti luettavaa, sen tulee olla mahdollisimman virheetöntä. Oikeinkirjoitukseen kuuluvat muun muassa yhdyssanat, lyhenneet ja pilkutus. (Waller 2011, 10.) Suomenkielisissä dokumentaatioissa tärkeää on myös vieraskielisten sanojen virheetön käyttäminen tekstissä.

Luvussa Dokumentoinnin hyödyt ja haitat mainittiin, miten akateemisesti vaikeaselkoiset dokumentaatiot haittaavat luettavuutta. Kielivirheettömyyteen voidaan laskea yksinkertaisen kielenkäyttö, jonka avulla kuka tahansa sisäistää dokumentaation informaation (Waller 2011, 8). Kappalejako on myös tärkeä osa luettavuutta. Mitä pidempi kappale on kyseessä, sitä todennäköisemmin lukija unohtaa kappaleen alun tekstin päästyään sen loppuun. (Schubert 2007.)

Dokumentaatiopohjat

Vaikka dokumenttipohjien käyttö voi tuntua aikaa vievältä tai turhalta, ne auttavat tekijää kirjoittamaan dokumentin tietyn formaatin mukaisesti. Jos dokumenttipohja on toimiva, lopputulos tulee näyttämään ammattimaiselta ja ennen kaikkea helposti luettavalta. On otettava huomioon, että jokainen projekti on erilainen ja siksi dokumenttipohja on muunnettava oman projektin tarpeiden mukaiseksi.

Isommissa yrityksissä dokumenttipohjat ovat tärkeä osa yritystä. Jos jokainen työntekijä tekisi dokumenttinsa omien sääntöjen mukaan, lukija joutuisi aina opettelemaan uuden tavan löytää tekstistä tarvitsemansa tiedon. Yrityksen mallin avulla hän tietäisi jo valmiiksi, mistä tarvittava tieto löytyy.

Useissa yrityksissä käytetään vielä Excel-taulukointityökalun pohjaa dokumentaatioissa aikataulusuunnitelmaa tehtäessä. Jos jokainen työntekijä tekee oman pohjansa, työnantajalle tulisi vaikeaksi hallita sitä, että työryhmän jäsenet ovat tehneet osuutensa ajallaan.

Internetissä käytössä ilmaisia tai maksullisia palveluja, joihin työntekijät voivat lisätä työtuntinsa ja työtehtävät. Ilmaisen palvelusta esimerkki on sivusto nimeltä Trello.com. Työnantaja pystyy näkemään palvelun kautta työntekijän päivitysten jälkeiset tilannetapahtumat.

6 YHTEENVETO

Aloittaessani tietotekniikan opiskelun ammattikorkeakoulussa ensimmäisenä päivänä luokanvalvojamme kertoi meille, millaista työmme tulisi olemaan. Hänen mukaansa työstämme hyvin pieni osuus koostuu ohjelmoinnista ja sen sijaan dokumentoiminen ja palaverikeskustelut veisivät suurimman osan ajastamme. Kouluvuosieni aikana olen oppinut, mitä hän tarkoitti.

Usein potentiaalinen sijoittaja tai työnantaja näkee ensimmäisenä projektista tehdyn dokumentaation. Jos dokumentti on huonosti toteutettu, jatkotoimenpiteet voidaan unohtaa. Silti useat yritykset näkevät dokumentoinnin turhana resurssikuluna. Jotta dokumentaatiot pysyisivät ajan tasalla, niillä tulisi olla omat vastuhenkilönsä, jotka vastaavat siitä, että yrityksellä olisi oma dokumentointitapansa, jota kaikki yrityksen jäsenet noudattavat.

Pelialan yrityksille ei ole vielä kehittynyt yhteistä dokumentointistandardia. Tämän vuoksi yhdenmukaisen tiedon löytäminen aiheesta on haastavaa. Täydellistä dokumentaatiota ei ole olemassa, mutta siitä voidaan tehdä selkeä panostamalla sekä dokumentin tekstiin että ulkoiseen näytävyyteen. Mitä enemmän dokumentoimisesta on kokemusta, sitä helpompaa ja nopeampaa sen kehittäminen jatkossa on.

Työn tarkoituksena oli tehdä tilaajalle pelisuunnittelun tärkeimpiin dokumentaatiotyyppien johdannosta. Opinnäytetyön tietoja ehditty hyödyntää tämän projektin aikana. Opinnäytetyötä voidaan käyttää perehdytykseen dokumentoinnista uusille työntekijöille.

Pelialan dokumentoinnissa pitää tietää myös pelisuunnittelusta, jota ei käyty tässä opinnäytetyössä tarkasti. Pelisuunnittelusta ja dokumentoinnista löytyy tietoa esimerkiksi tässä opinnäytetyössä lähteenä käytetystä Tony Mannisen kirjasta Pelisuunnittelijan käsikirja: Ideasta eteenpäin.

LÄHTEET

Apperley, Thomas H. 2006. Genre and game studies: Toward a critical approach to video game genres. Väitöskirja. University of Melbourne. Saatavissa: http://www.academia.edu/385966/Genre_and_Game_Studies_Toward_a_Critical_Approach_to_Video_Game_Genres. Hakupäivä 5.12.2013.

Berger, Andreas 2010. Technical Documentation Standard for Software development. Specific-Group. Saatavissa: http://www.specific-group.com/files/Technical_Documentation_StandardsV1.3.pdf. Hakupäivä 5.12.2013.

Bethke, Erik 2003. Game Development and production. USA, Texas: Wordware publishing, Inc.

Capri, Steve 2010. Developing Successful Software Documentation. Software-CEO. Saatavissa: http://www.softwareceo.com/downloads/file/sceo/white_papers/DevelopingSuccessfulSoftwareDocumentation.pdf. Hakupäivä 27.11.2013.

Chemuturi, Murali 2013. Mastering IT Project Management. USA: J. Ross Publishing.

Freeman, Tzvi 1997. Creating A Great Design Document. Gamasutra. Saatavissa: http://www.gamasutra.com/view/feature/3224/creating_a_great_design_document.php. Hakupäivä 27.11.2013.

Green, Michael 2011. One Page Proposal. Saatavissa: <http://www.gamedesigncenter.org/one-page-proposal>. Hakupäivä 4.12.2013.

Hackett, Scott 2007. How to Write an Effective Design Document. Saatavissa: <http://blog.slickedit.com/2007/05/how-to-write-an-effective-design-document/>. Hakupäivä 4.12.2013.

Hiltunen, KooPee – Latva, Suvi – Kaleva, Jari-Pekka 2013. Peliteollisuus - kehityspolku. Tekes. Katsaus 303/2013. Saatavissa:

http://www.tekes.fi/Julkaisut/peliteollisuus_kehityspolku.pdf. Hakupäivä

5.12.2013.

Hirschlein, Daniel 2010. 5 Types of Comments to Avoid Making in Your Code. RepeatGeek. Saatavissa: <http://repeatgeek.com/career/5-types-of-comments-to-avoid-making-in-your-code/>.

Hakupäivä 5.12.2013.

ITL Education Solutions Limited. Introduction to Information Technology. 2011 India: Pearson Education India.

Japenga, Robert. How to write a software requirements specification. MicroTools Inc. Saatavissa: <http://www.microtoolsinc.com/Howsrs.php>. Hakupäivä

5.12.2013.

Kuntz, Mare 2013. Developing Your Game Concept By Making A Design Document. GameDev. Saatavissa:

http://www.gamedev.net/page/resources/_/creative/game-design/developing-your-game-concept-by-making-a-design-document-r3004. Hakupäivä

4.12.2013.

Manninen, Tony 2007. Pelisuunnittelun käsikirja: Ideasta eteenpäin. Tallinna: Rajalla.

Maxwell Chandler, Heather 2013. The Game Production Handbook, 3rd Edition. Burlington, USA: Jones & Bartlett Learning.

McGuire, Morgan – Jenkins, Odest Chadwicke 2009. Book Extract: Creating Games: Mechanics, Content, and Technology. Game Career Guide. Saatavissa:

http://www.gamecareerguide.com/features/747/book_extract_creating_games_.php?page=1. Hakupäivä 5.12.2013.

Pickthall, Jason 2012. Just what is concept art?. Saatavissa:

<http://www.creativebloq.com/career/what-concept-art-11121155>. Hakupäivä

5.12.2013.

- Ramsay, Radolph 2010. Original BioShock pitch document released. GameSpot. Saatavissa: <http://www.gamespot.com/articles/original-bioshock-pitch-document-released/1100-6263197/>. Hakupäivä 4.12.2013.
- Rollings, Andrew – Adams, Ernest 2003. Andrew Rollings and Ernest Adams on Game Design. USA: New Riders Publishing.
- Rouse, Richard 2005. Game Design:Theory & Practice, Second Edition. USA: Wordware Publishing.
- Ryan, Tim 1999. The Anatomy of a Design Document, Part 1: Documentation Guidelines for the Game Concept and Proposal. Gamasutra. Saatavissa: http://www.gamasutra.com/view/feature/3384/the_anatomy_of_a_design_document_.php. Hakupäivä 27.11.2013.
- Schubert, Damion 2007. Writing Great Design Documents. GDCVault. Powerpoint-diaesitys. Saatavissa: <http://www.gdcvault.com/play/580/Writing-Great-Design>. Hakupäivä 27.11.2013.
- Sikora, Michael 2002. Removing The 'Tech' From 'Design Document'. Gamedev. Saatavissa: http://www.gamedev.net/page/resources/_/creative/game-design/removing-the-tech-from-design-document-r1764. Hakupäivä 5.12.2013.
- Silander, Simo. Dokumentointia. Saatavissa: http://cs.stadia.fi/~silas/ohjelmointi/c_opas-Dokument.html. Hakupäivä 27.11.2013.
- Spasov, Ivan 2013. Documentation for Indie Studios: Why do you need it?. Gamedev. Saatavissa: http://www.gamedev.net/page/resources/_/creative/game-design/documentation-for-indie-studios-why-do-you-need-it-r3111. Hakupäivä 5.12.2013.
- Silverster, Tynan 2013. Designing Games. CA, Sebastopol: O'Reilly Media, Inc.

Tervakari, Anne-Maritta 2008. Muita dokumentteja. Tampereen teknillisen yliopisto. Saatavissa: <http://hlab.ee.tut.fi/hmopetus/vpsist-oppimateriaali/5-dokumentointia/5-7-muita-dokumentteja>. Hakupäivä 4.12.2013.

Tervakari, Anne-Maritta 2011. Dokumentointia. Hypermedian opetus. Saatavissa: <http://hlab.ee.tut.fi/hmopetus/vpsist-oppimateriaali/5-dokumentointia>. Hakupäivä 27.11.2013.

Varanki, Hannele 2010. Aikataulut. Saatavissa: <http://hlab.ee.tut.fi/hmopetus/aikatauluttaminen>. Hakupäivä 5.12.2013.

Versiohallinta ja ohjelmiston dokumentointi 2012. Suomen yrittäjäopisto. Saatavissa: <http://www.syo.fi/tietotekniikka/erilliset-koulutusmoduulit/ohjelmointi/versiohallinta-ja-ohjelmiston-dokumentointi>. Hakupäivä 27.11.2013.

Vuori, Matti 2010. 125 pointtia dokumentoinnista. Saatavissa: http://www.mattivuori.net/julkaisuluettelo/liitteet/satavartti_pointtia_dokumentoinnista.pdf. Hakupäivä 5.12.2013.

Waller, Rob 2011. What makes a good document? - The criteria we use. University of Reading. Saatavissa: <http://www.reading.ac.uk/web/FILES/simplification/SC2CriteriaGoodDoc-7.pdf>. Hakupäivä 19.1.2014.

Wilson, Greg 2006. Off With Their HUDs!: Rethinking the Heads-Up Display in Console Game Design. Gamasutra. Saatavissa: http://www.gamasutra.com/view/feature/130948/off_with_their_huds_rethinking_.php?page=1. Hakupäivä 5.12.2013.